UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/756,894	01/13/2004	John Randall West	07464-0002	3627

25213 7590 02/26/2008

HELLER EHRMAN LLP
275 MIDDLEFIELD ROAD
MENLO PARK, CA 94025-3506

EXAMINER	
DAO, THUY CHAN	
ART UNIT	PAPER NUMBER
2192	

MAIL DATE	DELIVERY MODE
02/26/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/756,894	WEST ET AL.	
	Examiner	Art Unit	
	Thuy Dao	2192	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *03* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 November 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-48* is/are pending in the application.

4a) Of the above claim(s) *49* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-48* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *13 January 2004* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114.

Applicant's submission filed on November 29, 2007 has been entered.

2. Claims 1-48 have been examined.

Response to Amendments

3. Per Applicants' request, claims 1, 6, 38, and 43 have been amended and claim 49 has been canceled.

4. The objection to the specification and claims 1 and 6 is withdrawn in view of Applicants' amendments.

5. The 35 USC §101 rejection over claim 49 is withdrawn in view of Applicants' amendments.

Drawings

6. The drawings are objected to because minor informalities:

Figure 3: hand-written "32" and "Flow Control Structures";

Figure 4: in the specification, paragraph [0043] sets forth "an identification field 56 within the application metadata repository 54". However, the newly added identification filed "56" in Figure 4 is independent/separate with the application metadata repository "54" (i.e., not within the repository 54). For consistency, "56" should be deleted in both Figure 4 and specification; and in FIG. 4, "Identification Filed" should be deleted (i.e., only a single block for "Application Meta Data Repository" 54);

Figure 5: hand-drawn blocks; and block "Read Input Scripts" should have a rectangle shape;

Figure 6: a clean version without "54" should be provided;

Figure 7: a clean version should be provided; and

Figure 11: hand-written reference number "318".

Appropriate correction is required.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

Claim Objections

7. Claims 1 and 38 are objected to because of minor informalities.

Claim 1:

In light of the instant amendments and FIG. 10, [0049]-[0053], claim 1 is considered to read as:

- -A method for [[transforming test cases that are converted to an abstract representation and storing abstract representation of test cases into a database system]] <u>generating test scripts</u>, comprising:

importing test cases written in one or more scripting languages;

using semantic analysis to convert test cases to an abstract representation that includes application state, external interaction sequences and input data, wherein the application state is a set of application objects associated with a set of

attributes and their values, or represents a runtime snapshot of an application under test which defines a context of external interaction,

[[using environment mappings to provide platform independence of test case and the abstract representation providing a platform independent representation of test case;]]

storing abstract representation of test cases into a database system;

using mapping mechanism and the stored abstract representation for providing a platform independent representation of test cases; and

generating test scripts for multiple test execution environments- -.

Claim 38:

Claim 38 is considered to read as:

- -A system for [[transforming test cases to an abstract representation that are stored in a database]] generating test scripts, comprising:

a processor for ...;

logic that includes ..., the logic using [[environment mappings]] mapping mechanism and the stored abstract representation for providing platform independence of test cases and [[test scripts are generated for multiple test execution environments]]

logic that generates test scripts for multiple test execution environments- -.

Response to Arguments

8. Applicants' arguments have been fully considered.

a) "WinRunner creates tool dependency and reduces ability to share the tests" (Remarks, page 11, lines 13-14):

It appears that the Applicants argued about newly added limitations:

"*using environment mappings to provide platform independence of test cases and the abstract representation providing a platform independent representation of test cases; and*

generating test scripts for multiple test execution environments".

Applicants' arguments have been fully considered and are persuasive. After further consideration, a new ground of rejection has been applied in details below.

b) "Additionally, WinRunner does not use semantic analysis to convert test cases to abstract representations" (Remarks, page 11, lines 14-15):

The examiner respectfully disagrees. WinRunner explicitly discloses:

importing test cases written in one or more scripting languages (e.g., pp. 34-38, Recording Tests; page 40, Stop Recording and Save);

using semantic analysis to convert test cases to an abstract representation (e.g., pp. 42-43, GUI objects with attributes and values; pp. 22-26, converting recorded tests to sequences of interactions and GUI map with objects, attributes, and values).

c) "...test data is not separated from test steps in abstract representations" (Remarks, page 11, lines 15-16):

In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., "test data is not separated from test steps ...") are not recited in the rejected claim(s). Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See In re Van Geuns, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

In this Office action, the examiner establishes a new ground of rejections over WinRunner in view of Melamed as applied in details below.

Claim Rejections – 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be

patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over WinRunner (art of record, "WinRunner 7.0 Tutorial") in view of Melamed (art of record, US Patent Publication No. 2004/0107415 A1).

Claim 1:

WinRunner discloses *a method for transforming test cases that are converted to an abstract representation and storing abstract representation of test cases into a database system, comprising:*

importing test cases written in one or more scripting languages (e.g., pp. 34-38, Recording Tests; page 40, Stop Recording and Save)*;*

using semantic analysis to convert test cases to an abstract representation that includes application state, wherein the application state is a set of application objects associated with a set of attributes and their values (e.g., pp. 42-43, GUI objects with attributes and values; pp. 22-26, converting recorded tests to sequences of interactions and GUI map with objects, attributes, and values),

external interaction sequences and input data (e.g., page 35, Choosing a record mode; pp. 38-40, Recording a context sensitive test; pp. 44-46, Recording in analog mode; pp. 11-12, performing operations on applications under tests),

or represents a runtime snapshot of an application under test which defines a context of external interaction (e.g., pp. 48-50, Running the test including sequences of actions, which provides a runtime snapshot of application under test)*; and*

storing abstract representation of test cases into a database system (e.g., pp. 51-55, Test results; page 184, Maintain your test scripts; page 166, Test result window with test results; pp. 84-86, Running an existing test on a new version of an application under test; pp. 11-12, reusable tests).

WinRunner does not explicitly disclose *using environment mappings to provide platform independence of test cases and the abstract representation providing a platform independent representation of test cases; and generating test scripts for multiple test execution environments.*

However, in an analogous art, Melamed further discloses:

using mapping mechanism and the abstract representation for providing a platform independent representation of test cases (e.g., FIG. 18, [0085]-[0092]; [0071]); *and*

generating test scripts for multiple test execution environments (e.g., [0016], [0049], [0071], [0076]; FIG. 19, [0052]-[0054]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so to provide a unified multi-user and multi-platform testing system as suggested by Melamend (e.g., [0012]-[0017], [0020], [0071]).

Claim 2:

The rejection of claim 1 is incorporated. WinRunner also discloses *an application state represents a runtime snapshot of application under test, which defines the context of external interaction* (e.g., page 40, pp. 22-26).

Claim 3:

The rejection of claim 2 is incorporated. WinRunner also discloses *the application state includes a set of application objects, its attributes and attribute values* (e.g., pp. 34-38).

Claim 4:

The rejection of claim 2 is incorporated. WinRunner also discloses *the applications states corresponding to a test case are arranged in a hierarchical manner* (e.g., page 35, pp.38-40).

Claim 5:

The rejection of claim 2 is incorporated. WinRunner also discloses *the database system is a relational database management system* (e.g., pp. 11-12).

Claim 6:

The rejection of claim 2 is incorporated. Melamed further discloses *the database system is an XML database management system* (e.g., [0016], [0071]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 7:

The rejection of claim 2 is incorporated. WinRunner also discloses *the scripting languages can be typed or untyped programming languages used for recording or authoring test cases* (e.g., pp. 48-50).

Claim 8:

The rejection of claim 2 is incorporated. WinRunner also discloses *the external interaction sequences represent events invoked by external agents on the application objects* (e.g., pp. 11-12, page 166).

Claim 9:

The rejection of claim 8 is incorporated. WinRunner also discloses *the external agents can be either human agents or other software agents* (e.g., pp. 51-55).

Claim 10:

The rejection of claim 8 is incorporated. WinRunner also discloses *the interaction sequencing includes flow control structures for capturing sequential, concurrent, looping and conditional interactions* (e.g., page 184, pp. 84-86).

Claim 11:

The rejection of claim 2 is incorporated. WinRunner also discloses *implementing a syntax analyzer for incoming scripts* (e.g., page 166, pp. 44-46).

Claim 12:

The rejection of claim 11 is incorporated. WinRunner also discloses *the syntax analyzer is implemented one for each scripting language* (e.g., pp. 34-38, page 40).

Claim 13:

The rejection of claim 12 is incorporated. Melamed further discloses *the syntax analyzer utilizes rules of syntax analysis that are specified in Extended Backus-Naur Form (EBNF)* (e.g., [0016], [0051]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 14:

The rejection of claim 12 is incorporated. Melamed further discloses *the syntax analyzer generates a parse tree in the form of an Abstract Syntax Tree (AST)* (e.g., 0054], [0076]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 15:

The rejection of claim 2 is incorporated. Melamed further discloses *implementing a semantic analysis that converts an abstract syntax tree to an abstract test case representation based on an Application Object Model (AOM)* (e.g., [0016], [0051]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 16:

The rejection of claim 15 is incorporated. Melamed further discloses *the semantic analysis decomposes the test cases represented as an Abstract Syntax Tree into application state, external interaction sequences and input data* (e.g., [0072-0074]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 17:

The rejection of claim 15 is incorporated. Melamed further discloses *an application object model is a metadata representation for modeling application under test* (e.g., [0016], [0051]).

Claim 18:

The rejection of claim 17 is incorporated. Melamed further discloses *the metadata representation includes object type definitions for application objects* (e.g., [0054], [0076]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 19:

The rejection of claim 17 is incorporated. Melamed further discloses *the metadata representation includes attribute definitions for each application object type* (e.g., [0016], [0071]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 20:

The rejection of claim 17 is incorporated. Melamed further discloses *the metadata representation includes definition of methods and events that are supported by each application object type* (e.g., [0014], [0018]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 21:

The rejection of claim 17 is incorporated. Melamed further discloses *the metadata representation includes definition of effects of events on an application state* (e.g., [0093], [0097]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 22:

The rejection of claim 18 is incorporated. Melamed further discloses *application object type definitions include additional categorization of each application object types into hierarchical, container and simple types* (e.g., [0093], [0014], [0018]).

Claim 23:

The rejection of claim 22 is incorporated. Melamed further discloses *the hierarchical object types are associated with an application state of its own; wherein application object types that can contain instances of other objects are termed as container types* (e.g., [0021], [0010], [0040]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 24:

The rejection of claim 23 is incorporated. Melamed further discloses *the state associated with a hierarchical application object type is a modal application state or a nonmodal application state* (e.g., [0009], [0013], [0054]).

Claim 25:

The rejection of claim 24 is incorporated. Melamed further discloses *a modal application state restricts possible interactions to application object instances available within the current application state* (e.g., [0085-0086]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 26:

The rejection of claim 22 is incorporated. Melamed further discloses *the effects of events on an application state capture one or more consequences of the event to the application state* (e.g., [0065], [0079]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 27:

The rejection of claim 26 is incorporated. Melamed further discloses *a consequence of an event is selected from, creation of a new object instance of a given type, deletion of an object instance of a given type, modification of attributes of an*

existing object instance and selection of an instance of an object type (e.g., [0093], [0018]).

Claim 28:

The rejection of claim 27 is incorporated. Melamed further discloses *creation of a new instance of an object of type that is hierarchical results in creation of a new application state* (e.g., [0054-0056]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 29:

The rejection of claim 27 is incorporated. Melamed further discloses *selection of an object instance of type that is hierarchical results in selection of the application state associated with that object instance* (e.g., [0065], [0079]).

Claim 30:

The rejection of claim 2 is incorporated. WinRunner also discloses *enriching the abstract representation of test cases with information from an application metadata repository* (e.g., pp. 42-43, pp. 22-26).

Claim 31:

The rejection of claim 30 is incorporated. WinRunner also discloses *the enrichment of abstraction representation of test cases involves extracting values for those attributes of application objects associated with the test cases that are missing in the incoming test scripts* (e.g., pp. 48-50).

Claim 32:

The rejection of claim 30 is incorporated. WinRunner also discloses *enriching the abstraction representation of test cases includes decoupling of test cases from their recording or authoring environments* (e.g., pp. 51-55).

Claim 33:

The rejection of claim 30 is incorporated. WinRunner also discloses *enriching the abstraction representation of test cases allows usage of attributes that are stable within an application metadata representation* (e.g., pp. 42-43, page 35).

Claim 34:

The rejection of claim 33 is incorporated. WinRunner also discloses *using an identification field for a given object within the application metadata repository improves the reusability of a test case instead of a label used to represent the same object within a user interface which can change based on the locale of the application* (e.g., pp. 38-40, pp. 44-46).

Claim 35:

The rejection of claim 33 is incorporated. WinRunner also discloses *using an identification field allows to overcome the problem of different test execution environments using different attributes to identify the same application object* (e.g., page 184, pp. 84-86).

Claim 36:

The rejection of claim 35 is incorporated. WinRunner also discloses *enriching the abstraction representation of test cases enables representation of test cases that are test execution environment independent* (e.g., pp. 11-12, pp. 51-55).

Claim 37:

The rejection of claim 2 is incorporated. WinRunner also discloses *separating application object attributes and input data from external interaction sequencing provides automatic parameterization* (e.g., page 40, pp. 44-46).

Claim 38:

WinRunner discloses *a system for transforming test cases to an abstract representation that are stored in a database, comprising:*

a processor for importing test cases written in one or more scripting languages (e.g., pp. 34-38, Recording Tests; page 40, Stop Recording and Save);

logic that includes semantic analysis for converting test cases to an abstract representation that includes application state (e.g., page 35, Choosing a record mode; pp. 38-40, Recording a context sensitive test; pp. 44-46, Recording in analog mode; pp. 11-12),

external interaction sequences and input data (e.g., pp. 42-43, GUI objects with attributes and values; pp. 22-26, GUI map with objects, attributes, and values); *and*

a database that stores abstract representation of test cases, the application state being a set of application objects associated with a set of attributes and their values (e.g., pp. 51-55, Test results; page 184, Maintain your test scripts; page 166, Test result window with test results; pp. 84-86, Running an existing test on a new version of an application under test),

or represents a runtime snapshot of an application under test which defines a context of external interaction (e.g., pp. 48-50, Running the test including sequences of actions, which provides a runtime snapshot of application under test).

WinRunner does not explicitly disclose *logic using environment mappings providing platform independence of test cases and logic that generates test scripts for multiple test execution environments.*

However, Melamed further discloses:

logic using mapping mechanism and the stored abstract representation for providing platform independence of test cases (e.g., FIG. 18, [0085]-[0092]; [0071]) *and*

logic that generates test scripts for multiple test execution environments (e.g., [0016], [0049], [0071], [0076]; FIG. 19, [0052]-[0054]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so to provide a unified multi-user and multi-platform testing system as suggested by Melamend (e.g., [0012]-[0017], [0020], [0071]).

Claim 39:

The rejection of claim 38 is incorporated. WinRunner also discloses *a syntax analyzer for incoming scripts* (e.g., [0184], [0166], [0084]-[0086]).

Claim 40:

The rejection of claim 38 is incorporated. Melamed further discloses *logic for implementing a semantic analysis that converts the abstract syntax tree to an abstract test case representation based on an Application Object Model (AOM)* (e.g., [0016], [0071]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 41:

The rejection of claim 38 is incorporated. WinRunner also discloses *logic for selecting an object instance of type that is hierarchical results in selection of the application state associated with that object instance* (e.g., [0042—[0043]; [0022]-[0026]).

Claim 42:

The rejection of claim 38 is incorporated. Melamed further discloses *logic for enriching the abstraction representation of test cases to enable representation of test cases that are test execution environment independent* (e.g., [0013], [0054]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 43:

WinRunner discloses a *computer system for storing abstract representations of test cases in a database, comprising:*

a processor; a memory coupled to the processor (e.g., pp. 142-144),

the memory storing program instructions executable by the processor for converting test cases to an abstract representation that includes application state (e.g., pp. 34-38, Recording Tests; page 40, Stop Recording and Save; pp. 11-12),

external interaction sequences and input data (e.g., pp. 42-43, GUI objects with attributes and values; pp. 22-26, GUI map with objects, attributes, and values),

the application state being a set of application objects associated with a set of attributes and their values (e.g., page 35, Choosing a record mode; pp. 38-40, Recording a context sensitive test; pp. 44-46, Recording in analog mode),

or represents a runtime snapshot of an application under test which defines a context of external interaction (e.g., pp. 48-50, Running the test including sequences of actions, which provides a runtime snapshot of application under test)*; and*

a database that stores abstract representation of test cases (e.g., pp. 51-55, Test results; page 184, Maintain your test scripts; page 166, Test result window with test results; pp. 84-86, Running an existing test on a new version of an application under test).

WinRunner does not explicitly disclose *environment mappings provide platform independence of test case and the abstract representation provides a platform*

independent representation of test case, and wherein test scripts are generated for multiple test execution environments.

However, Melamed further discloses:

environment mappings provide platform independence of test case and the abstract representation provides a platform independent representation of test case (e.g., FIG. 18, [0085]-[0092]; [0071]), *and*

wherein test scripts are generated for multiple test execution environments (e.g., [0016], [0049], [0071], [0076]; FIG. 19, [0052]-[0054]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so to provide a unified multi-user and multi-platform testing system as suggested by Melamend (e.g., [0012]-[0017], [0020], [0071]).

Claim 44:

The rejection of claim 43 is incorporated. WinRunner also discloses *a syntax analyzer for incoming scripts* (e.g., [0011]-[0012], [0051]-[0055]).

Claim 45:

The rejection of claim 44 is incorporated. Melamed further discloses *the syntax analyzer generates a parse tree in the form of an Abstract Syntax Tree (AST)* (e.g., [0054], [0076]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 46:

The rejection of claim 45 is incorporated. Melamed further discloses *logic to implement semantic analysis and convert the AST to an abstract test case representation based on an Application Object Model (AOM)* (e.g., [0016], [0071]).

It would have been obvious to a person having ordinary skill in the art at the time the invention was made to combine Melamed's teaching into WinRunner's teaching. One would have been motivated to do so as set forth above.

Claim 47:

The rejection of claim 43 is incorporated. WinRunner also discloses *logic for enriching the abstract test case representation with information from an application metadata repository* (e.g., [0034]-[0038], [0040]).

Claim 48:

The rejection of claim 43 is incorporated. WinRunner also discloses *logic for separating application object attributes and input data from external interaction sequencing to provide automatic parameterization* (e.g.,[0044]-[0046]).

Conclusion

11. Any inquiry concerning this communication should be directed to examiner Thuy Dao (Twee), whose telephone/fax numbers are (571) 272 8570 and (571) 273 8570, respectively. The examiner can normally be reached on every Tuesday, Thursday, and Friday from 6:00AM to 6:00PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam, can be reached at (571) 272 3695.

The fax phone number for the organization where this application or proceeding is assigned is (571) 273 8300.

Any inquiry of a general nature of relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is (571) 272 2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only.

For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

T. Dao

TUAN DAM
SUPERVISORY PATENT EXAMINER